International Journal of Applied Engineering and Technology ISSN: 2277-212X (Online)
An Open Access, Online International Journal Available at http://www.cibtech.org/jet.htm
2016 Vol. 6 (2) April-June, pp.1-7/Baktash et al.
Research Article

FORMAL DEVELOPMENT OF ONLINE SHOPPING SYSTEM USING EVENT-B

***Jawid Ahmad Baktash[1], Tomokazu Nagata[1] and Mursal Dawodi[2]**
[1]Graduate School of Engineering and Science, University of the Ryukyus, 1 Senbaru, Nishihara Okinawa Japan 903-0213
[2]School of Electrical Engineering and Computer Science, Technical University of Berlin, 10623 Berlin, Germany
**Author for Correspondence*

ABSTRACT
The goal of this paper is to explore the use of formal method for online shopping system. And it will deal with the design of shopping system, we are presenting different methodologies for message correctness of this system to make the employs abstract interpretation techniques for creating a trace based model for the system. The shopping system is a process of ordering and billing of departmental store which is located in various cities, nowadays one of the core research direction in a constantly growing distributed environment is the improvement of these systems, so the online shopping system can administrate the users and customers. Online shopping system is beneficial for all customers as well as merchants. Nowadays people like to buy things from online stores rather than virtual markets because of their hectic routines. So, the entails interaction between the customers and the owners are dealing with buying, selling of commodities, as a result the code for online shopping system generated and used to apply all authentication rules. Also it would make searching, viewing and selection of a product easier. It contains a sophisticated search engine for users to search for products specific to their needs. The user can view the complete specification of each product. The core technology seems ready for prime time and government projects deploy it to the masses. But application on issues may hamper online billing system adoption for online applications.

Keywords: *Online Shopping System, Event-B, Formal Method, Message Ordering*

INTRODUCTION
The new concept of impose and spawned session that we interdicted led our research to original statement about the intruders knowledge and online shopping system code with respect to the security, in most developing countries and (Afghanistan) selling and buying of goods are done in market. The concept of exchange leads to the concept of market. This situation is faced with diverse problems, E-shopping and online shopping represents the direction of future trade development. Promotion of E-System will highly effect society and economy.
According to a simple market research, there are large domestic markets chains located around every city. The problems include the need for buyers and sellers to come into physical contact, the necessary need for potential buyers to visit the shop and non-flexibility in time usage. In over to overcome these problems, an online shopping system is developed.

Formal Method
Compelling evidence of correctness early enough to be useful, cheaply enough to be feasible, and on the basis of modeling that is simple enough to be credible. Formal specifications feature a high degree of logical precision which eliminates much of the ambiguity that is found inevitably in informal specifications (Baktash *et al.,* 2015).
Formal proofs eliminate ambiguity and subjectivity from requirements analysis by providing a logical and precise argument for the behavior of the requirements. This enhances the analysis performed in informal reviews and inspections. The use of formal specifications and formal proofs provides a systematic, repeatable approach to analysis. The use of formal specifications and proofs are not an all-or-nothing

approach. It can be tailored to the level of rigor appropriate to a given budget, schedule, and technical need.

That is, it can be scaled to match the needs of a project. Formal specifications and proofs can be applied at any life cycle phase, including early in the life cycle where better analysis approaches are currently most needed. Detecting and fixing defects earlier in the process is far cheaper than finding them later in the process.

For example, one could tailor the use of specifications and proofs to focus on the verification of critical properties early in the life cycle (Yadav and Butler, 2006). Formal specification and proofs can be supported by computer-based tools. This provides automation for tasks such as consistency checking and the preparation of proofs.

These tools are analogous to the use of automatic calculators (and computers) in the analysis of engineering equations, but rather than "plug in" numbers into a formal specification, one "plugs in" symbolic variables and calculates the equivalent of a closed form solution. This is an important benefit that provides an additional level of assurance as well as reducing the cost of certain aspects of the analysis. These tools greatly enhance the repeatability of the analysis by allowing proofs to be re-executed. This also allows quick answers to the consequences of "What if..." questions early in the developmental life cycle.

Formal specifications and proofs complement the existing testing approach, but go beyond what testing can accomplish. They make perfect testing by providing a precise specification from which better test plans can be derived. They go beyond testing because they have the unique capability to show that key properties are satisfied in entire classes of scenarios.

In summary, formal methods enable defects in requirements to be detected earlier than otherwise, and can greatly reduce the incidence of mistakes in interpreting, formalizing, and implementing correct requirements.

Furthermore, used early in the life cycle, formal methods yield formalized statements that can be analyzed and their consequences calculated in a repeatable manner. In addition to these generic benefits attributable to the full spectrum of formal methods, the most rigorous and fully formal versions of formal methods cause more defects to be detected than would otherwise be the case and, in certain circumstances, subject to certain caveats, guarantee the absence of certain defects. When used judiciously and skillfully on suitable applications, Abrial (1996).

Online Shopping System Use Case Model

In this context diagram, the information provided to and received from the 'Online Shopping' is identified.

The arrows represent the information received or generated by the application. The users can view complete specification, a customer is registering into the system accordingly, if the user name is already in the system the error will display, in this system we can follow bellow objectives,

1. Add register and maintain record of available products,
2. Add and maintain of customer details and description of new products,
3. Add and maintain new entered category of products, provide a convenient of billing pattern,
4. Provides financial reports to all the users and customers, Making an easy to use graphical user interface,
5. Add, Search, Order and cancel of the orders.

The Event-B is a formal technique consists of describing rigorously the problem, introduce solutions or details in the refinement steps to obtain more concrete specifications, and verifying that proposed solutions are correct.

The system is modeled in terms of an abstract state space using variables with set theoretic types and the events that modify state variables. Event-B, a variant of B, was designed for developing distributed systems.

In Event-B, the event consists of guarded actions occurring spontaneously rather than being invoked. The invariants state properties that must be satisfied by the variables and maintained by the activation of the

events. In the refinement steps, guards may be strengthened and the new events may be introduced. Abstract and concrete variables are related through gluing invariants. At each refinement step a more concrete specification of the system are obtained. This technique requires the discharge of the proof obligations for consistency checking and refinement checking (Naseri *et al.*, 2015).

- MACHINE: machine show the name of the program that is designed for a specific machine like (MACHINE : OBS)
- SETS : set is use to show the number of processes in machine like (Citizen, OBS Machine, OBS server)
- VARIABLES: In computer programming, a variable is a storage location and an associated symbolic name (an identifier) which contains some known or unknown quantity or information, a value. The variable name is the usual way to reference the stored value, this separation of name and content allows the name to be used independently of the exact information it represents.
- INVARIANT: an expression whose value doesn't change during program execution
- INITIALISATION: Initialization is the process of locating and using the defined values for variable data that is used by a computer program.

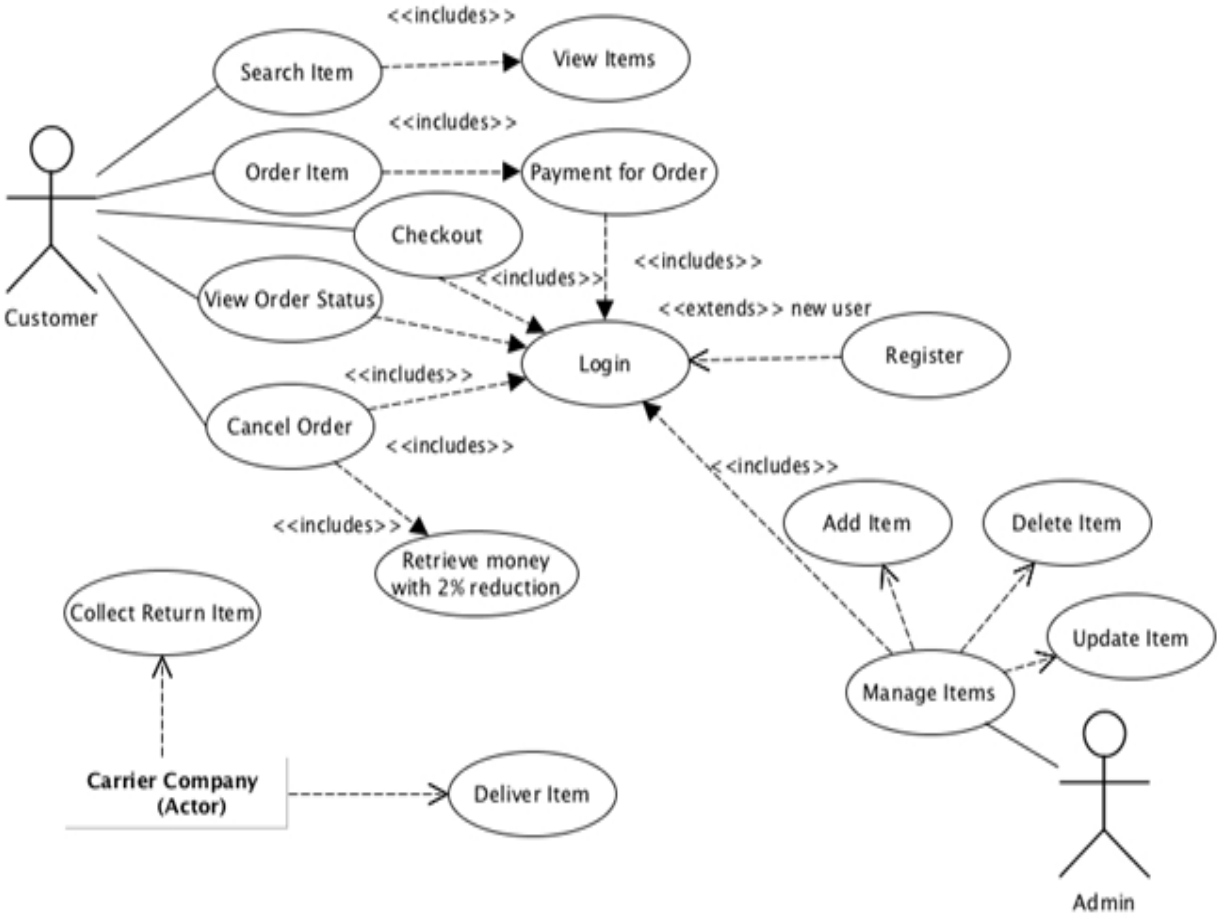

Figure 1: OSS Use Case Diagram which is a Graphic Depiction of the Interactions among the Elements of this System and it Represents the Responsibilities and Functionalities of the Stockholders

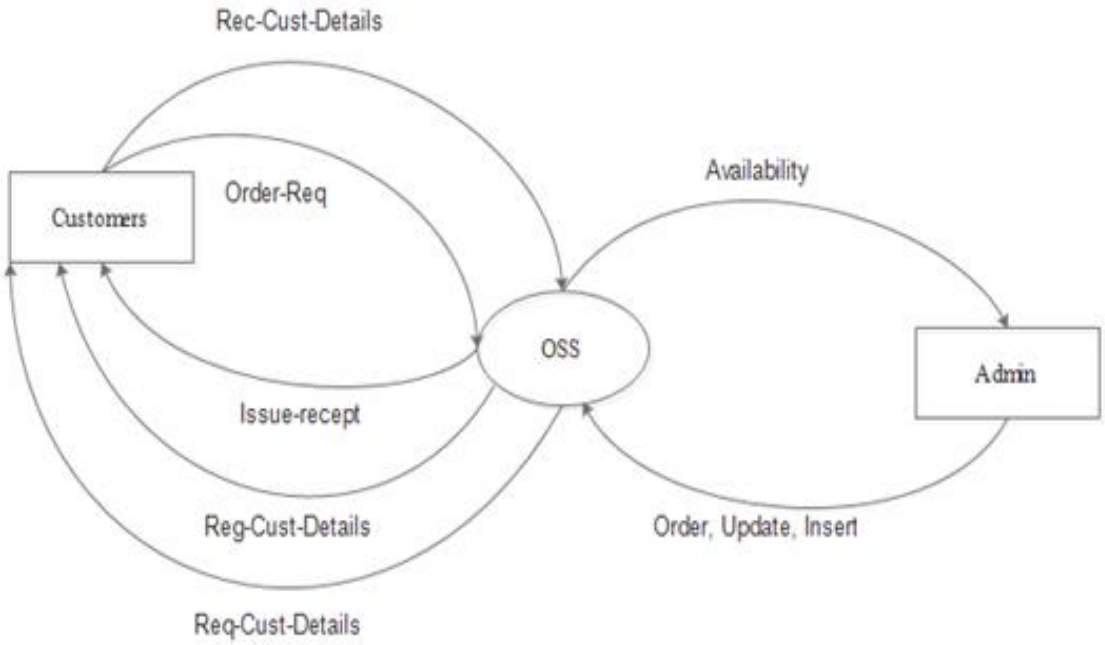

Figure 2: Presents the OSS Model, Describes the Process of Ordering Item(s)

For ordering of message broadcasting and multicasting with online Shopping system we use the FIFO (first in first order) ordering algorithm, with this algorithm when the first user is asking for the need the user has priority to get the response.

$$Customer[An] = customer[An] - 1\& .....$$

$$= customer[An] - (n-1) = customer[An] - n \wedge$$

$$customer[Am] \geq customer[Am] - 1\&$$

$$\forall m = \{1, 2, 3, ...n-1, n\} \quad m \neq (-m)$$

Activities are the particular operation of the system. We are using the activity diagram to construct the executable system by using forward and reverse engineering techniques.

Because it is possible explicitly to describe parallel events, so activity diagram is well suited for the illustration of running the processes for the online shopping system figure 3 shows the activity diagram if the system

Conclusion

We proposed a general formulation of Event-B model for Online Billing System. This can help Customers to use their time, money more effectively, OBS supports formal authentication where the law requires it, and may be the only mechanism to do so. OBS may therefore become an enabler for new online applications.

They simplify procedures that service providers are required to implement. We expect that OBS will blossom in these contexts, but not generally replace other authentication schemes. Indeed OBS is one of the remote shopping, anyone can buy/sell goods from his/her own PC, unlikely to waste time he/she can be the felicitated with ultra- confidentiality.

In addition OBS is highly preferable security based whereas, it is mathematically passed to system and EVENT-B is a programming language, which programs the formal method.

Machine :

$Online\ Shopping\ System(OSS)$

Sets:

$CUSTOMER = \{c1, c2, c3, ...cn\}$

$MESSAGE = \{m1, m2, m3, ...mn\}$

Variable:

$Logged\text{-}in, send, receive,$
$update, insert, delete$

Invariant:

$$Logged-in \in OSS(process) \wedge$$
$$Send \in Customer \in OSS \wedge$$
$$receive \in Customer \in OSS \wedge$$
$$update \wedge insert \wedge delete$$
$$\in OBS(process)$$

$$!(c,m).\begin{Bmatrix} c \in customer\ \& \\ m \in message \\ \&(m \mapsto message) \in c \\ \Rightarrow m \in dom(sender) \end{Bmatrix}$$

$$!(c,m).\begin{Bmatrix} c \in customer\ \& \\ m \in message \\ \&(m \mapsto c) \in receive \\ c \in receive(m) \\ \Rightarrow c \notin sender(m) \end{Bmatrix} */$$

Initialization

$(c \mapsto m) \in receive,\ THEN$
$regester := reg \cup \{c \mapsto m\};$
END

$Event : Insertion$
$C1, C2 \in Customer;$
$When$
$$\begin{bmatrix} C1 \in cust1 \\ C2 \in Cust2 \end{bmatrix} \wedge C1 \neq C2 \wedge$$
$$C1, C2 \in Cust$$
$Then$
$C1\&C2 : cust \rightarrow insert$
End

$Logged - in := \varnothing \,||\, send(m) := \varnothing$
$\wedge receive := \varnothing \wedge insert := \varnothing$
$\wedge update := \varnothing \wedge delete := \varnothing$
$Event : regester$
$c \in cust \wedge m \in mess$
$WHEN \quad m \in dom(send) \wedge$

$Event : Update$
$\begin{pmatrix} U : update \\ m : mess \end{pmatrix} =$
$When \quad m \notin dom(cust)$
$Then \quad send := send \cup$
$\{m \mapsto U\}$
$||\, cust := U \cup \{U \mapsto m\}$
$END;$
$Event : Delete$
$When$
$\quad m \in mess \wedge mess \in dom(cust)$
$\quad \& (cust \mapsto m) \notin delete;$
$Then$
$\quad cust := delete \cup \{record(m) \mapsto m\}$
$\quad End$
End

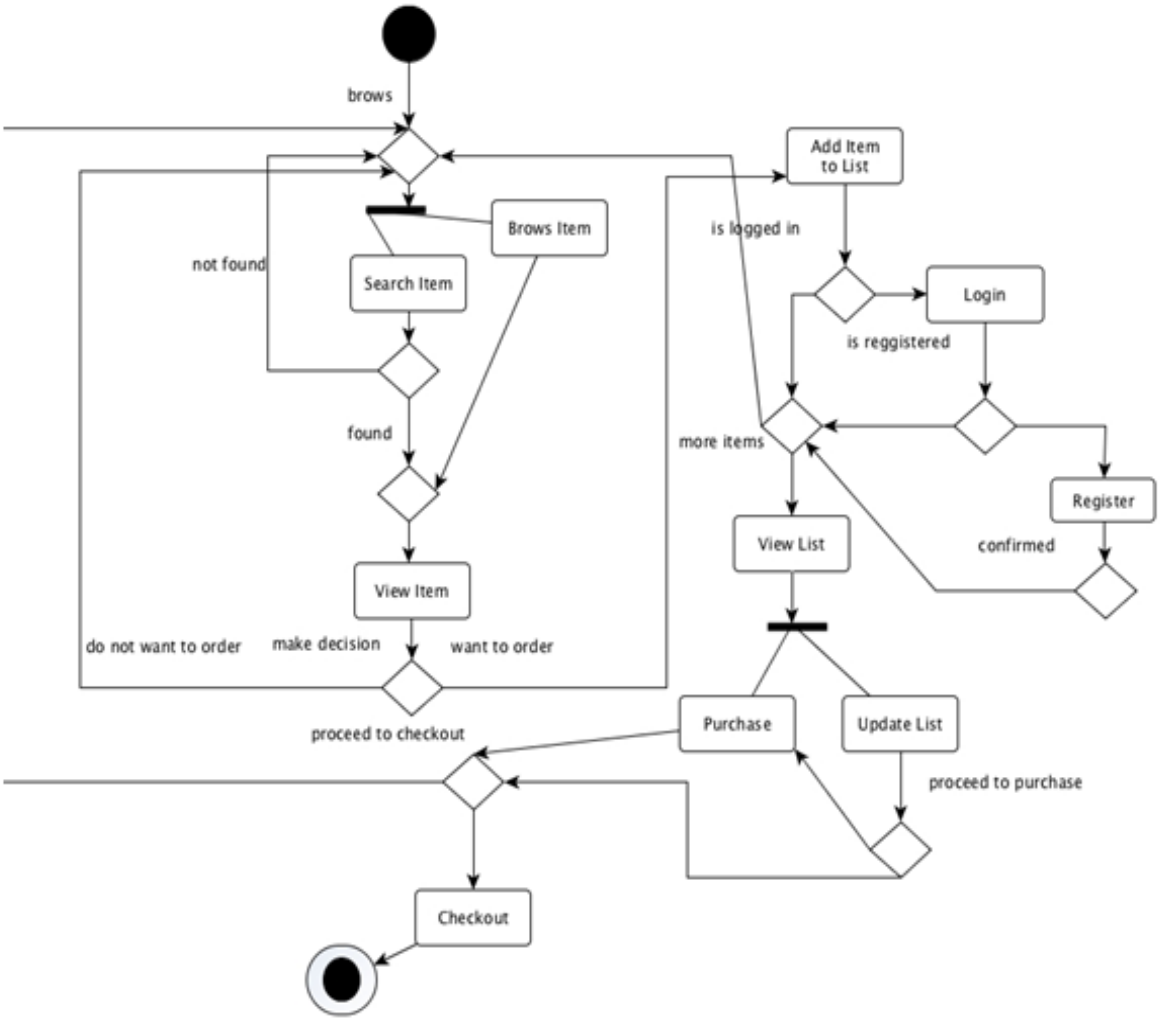

Figure 3: Activity Diagram, Represents Workflow for Purchasing of Item(s). Customer can Search Item(s) if it is Available in the System he/she can Add it/them to the List. Customer should be Registered and Logged in to the System as a Prerequisite to Insert/Update/Cancel the List and Purchase Item(s)

Conflict of Interests
The authors have no conflict of interests or financial ties regarding the publication of this paper.

REFERENCES
Baktash JA and Nagata T (2014). Formal Development of Electronic Identity Card System Using Event-B. *Information Journal* **17**(9B) 4677-4685 2014/9.
Baktash JA, Shiroma T, Nagata T, Taniguchi Y, Nakamura M and Dawodi M (2015). Formal Implementation of Routing Information Protocol using Event-B, *Information Journal* **18**(2) 705-718, 2015/2.
Naseri K, Baktash JA, Kang D and Nagata T (2015). Accessibility Assessment of Health Services in Kabul City, Using GIS. *Information Journal* **18**(11) 4643-4650, 2015/11.

Brands SA (2000). *Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy*, (MIT Press, Cambridge, England).
Abrial JR (1996). Extending B without changing it (for developing distributed systems). In H. Habrias, editor, *First B Conference*.
Yadav D and Butler M (2006). Rigorous design of fault-tolerant transactions for replicated database systems using Event B. In *Rigorous Development of Complex Fault-Tolerant Systems* 343-363 **4157** of Lecture Notes in Computer Science, (Springer-Verlag, Berlin Heidelberg, Germany).
Metayer C, Abrial JR and Voison L (2005). Event-B language. Technical Report, RODIN deliverables 3.2. Available: http://rodin.cs.ncl.ac.uk/deliverables/D7.pdf.
Spil TAM, Schuring RW and Katsma C (2002). Assessing Resistance of professionals as a determinant of IT-diffusion and IT-use in healthcare," presented at *9th European Conference on IT Evaluation (ECITE 2002)*, Paris, France.
Asharif F, Tamaki S, Nagado T, Nagata T and Asharif MR (2011). Analysis of MIMO System with Internal Model Controller Including Uncertainty of Time-Delay Element, External Disturbance and Feedback Noise. *Information Journal* **14**(1) 5-16 2011/1.